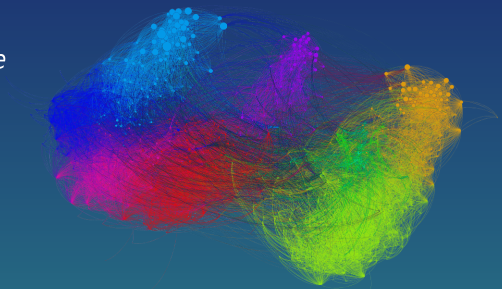
Graphs in Machine Learning

Michal Valko

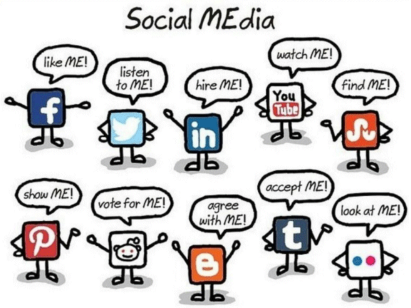
Inria Lille - Nord Europe, France

Partially based on material by: Andreas Krause, Branislav Kveton, Michael Kearns

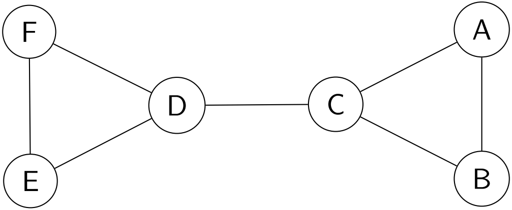


Graphs from **social networks**

- people and their interactions
- directed (Twitter) and undirected (Facebook)
- structure is rather a *phenomena*
- typical ML tasks
  - advertising
  - product placement
  - link prediction (PYMK)

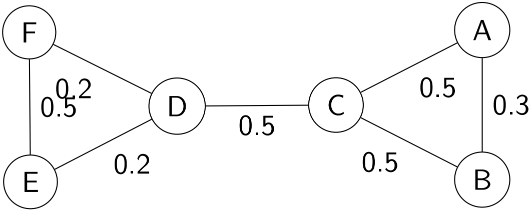


Success story #1 **Product placement** - problem



Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 **Product placement** - problem

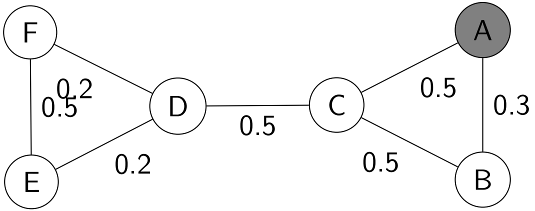


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem

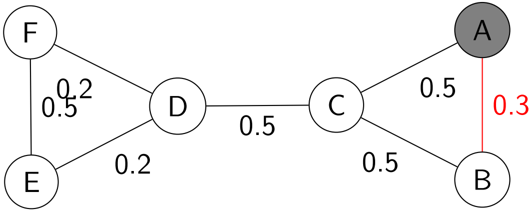


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem

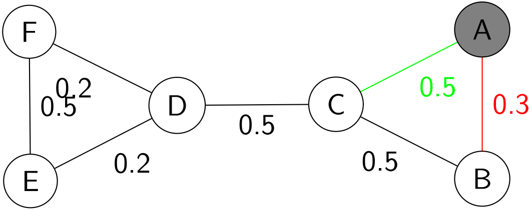


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 **Product placement** - problem

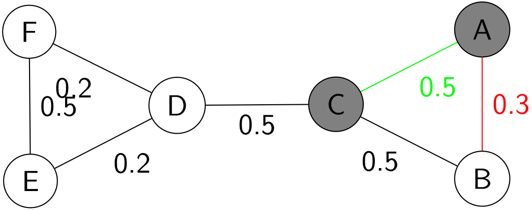


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem

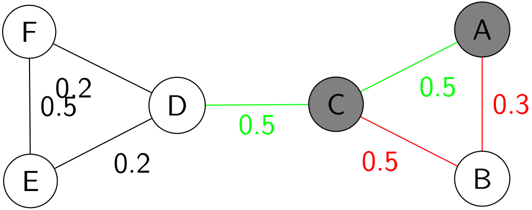


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem

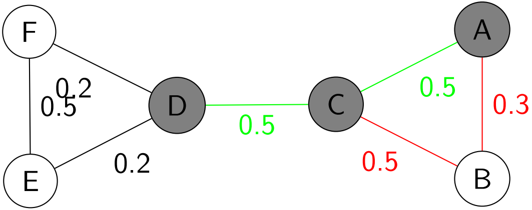


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 **Product placement** - problem

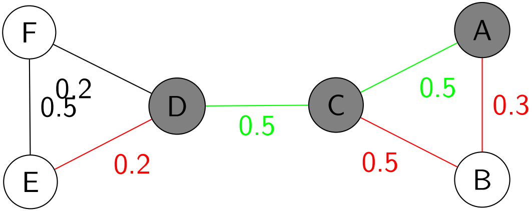


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem

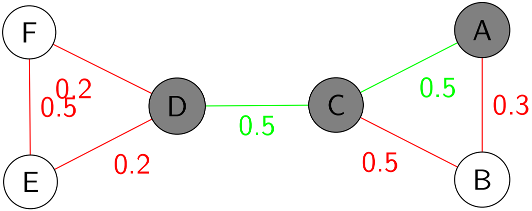


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 **Product placement** - problem



Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

$F(S) =$ Expected number of people influenced when targeting $S \subseteq V$ under some propagation model - e.g., cascades

How would you choose the target customers?

highest degree, close to the center, . . .

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Submodularity: Definition

A **set function** on a discrete set A is **submodular** if for any $S \subseteq T \subseteq A$ and for any $e \in A \setminus T$

$$f(S \cup \{e\}) - f(S) \geq f(T \cup \{e\}) - f(T)$$

Example: $S = \{\text{stuff}\} = \{\text{bread, apple, tomato, } \dots\}$
$f(V) = \text{cost of stuff to get } V$

$f(\{\text{bread}\}) = c(\text{bakery}) + c(\text{bread})$
$f(\{\text{bread}, \text{apple}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{apple}\})$
$f(\{\text{bread}, \text{tomato}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{tomato})$
$f(\{\text{bread}, \text{tomato}, \text{apple}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{tomato}) + c(\text{apple})$

Adding apple to the smaller set costs more!

$$\{\text{bread}\} \subseteq \{\text{bread}, \text{tomato}\}$$
$$f(\{\text{bread}, \text{apple}\}) - f(\{\text{bread}\}) > f(\{\text{bread}, \text{tomato}, \text{apple}\}) - f(\{\text{tomato}, \text{bread}\})$$

Diminishing returns: Buying in bulk is cheaper!

Submodularity: Application

Special case: f is also **nonnegative** and **monotone**.

Objective: Find $\arg\max_{S \subseteq A, |S| \leq k} f(S)$

Property: NP-hard in general

Other examples: information, graph cuts, covering, ...

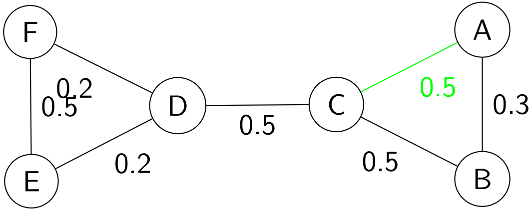
Link to our **product placement** problem on a **social network graph?**

submodular?, nonnegative?, monotone?, k?

Let $S^\star = \arg\max_{S \subseteq A, |S| \leq k} f(S)$ where f is monotonic and submodular set function and let S_{greedy} be a **greedy solution**.

$$\text{Then} \quad f(S_{\text{greedy}}) \geq \left(1 - \frac{1}{e}\right) \cdot f(S^\star).$$

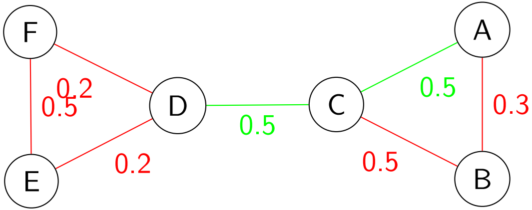
Success story #1 **Product placement** - solution



Key idea: Flip coins c in advance $\rightarrow$ "live" edges

(cf. Andreas Krause `http://submodularity.org/`)

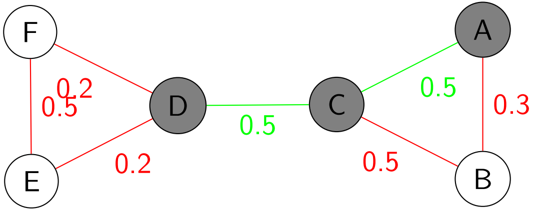
Success story #1 **Product placement** - solution



Key idea: Flip coins c in advance $\rightarrow$ "live" edges

(cf. Andreas Krause `http://submodularity.org/`)

Success story #1 **Product placement** - solution

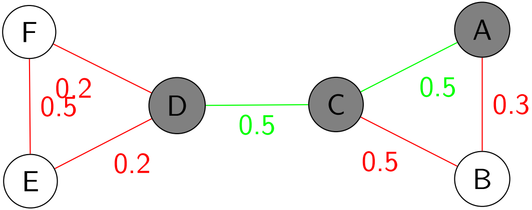


Key idea: Flip coins c in advance $\rightarrow$ "live" edges

$F_c(V)$ = People influenced under outcome c (set cover!)

(cf. Andreas Krause `http://submodularity.org/`)

Success story #1 **Product placement** - solution

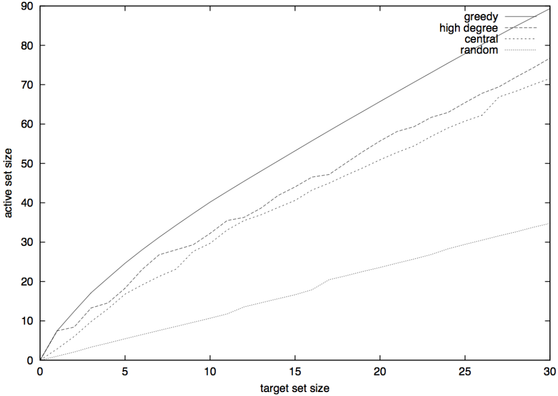


Key idea: Flip coins c in advance → "live" edges
$F_c(V)$ = People influenced under outcome c (set cover!)
$F(V) = \sum_c P(c)F_c(V)$ is submodular as well!

(cf. Andreas Krause `http://submodularity.org/`)

Success story #1 **Product placement** - comparison

propagation on the ArXiv/Physics co-authorship dataset



greedy approximation does better than the centrality measures

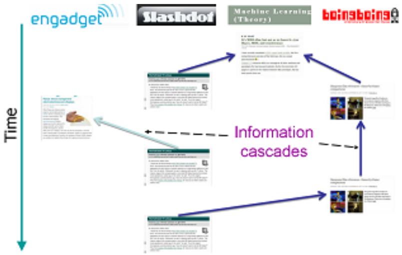
Graphs from **utility and technology networks**

- link services
- power grids, roads, transportation networks, Internet, sensor networks, water distribution networks
- structure is either *hand designed* or not
- typical ML tasks
  - best routing under unknown or variable costs
  - identify the node of interest

Berkeley's Floating Sensor Network

Graphs from **information networks**

- web
- blogs
- wikipedia
- typical ML tasks
  - find influential sources
  - search (pagerank)



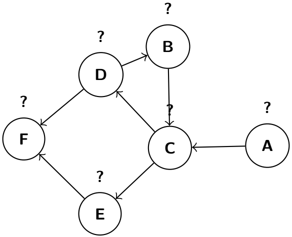
Blog cascades (ETH) - *submodularity*

Success story #2 **Google PageRank**

Objective: **Rank** all web pages (nodes on the graph) by how **many** other pages link to them and how **important** they are.

basic PageRank is independent of query and the page content

Internet $\rightarrow$ graph $\rightarrow$ matrix $\rightarrow$ stochastic matrix $\mathbf{M}$ $(\sum_j \mathbf{M}_{ij} = 1)$



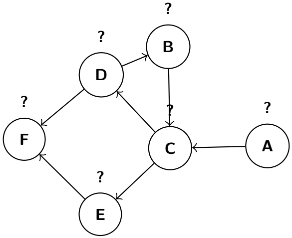
Success story #2 **Google** … **Rank**

Objective: **Rank** … ph) by how **many** other pages lin… …e.

Random Surfer Process

basic PageRan… …e content

Internet $\to$ graph… …$\mathbf{M}$ $(\sum_j \mathbf{M}_{ij} = 1)$



Success story #2 Google ... Rank

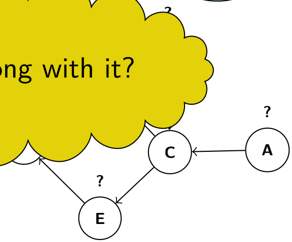
Objective: **Rank** ... ph) by how **many** other pages lin... e.

basic PageRan... e content

Internet → graph ... $\mathbf{M}$ $(\sum_j \mathbf{M}_{ij} = 1)$

Random Surfer Process

What is wrong with it?



Success story #2 **Google PageRank**

`http://infolab.stanford.edu/~backrub/google.html`:

PageRank can be thought of as a model of user behavior. We assume there is a "random surfer" who is given a web page at random and keeps clicking on links, never hitting "back" but eventually gets bored and starts on another random page.

- page is important if important pages link to it
  - circular definition
- importance of a page is distributed evenly
- probability of being bored is 15%

Success story #2 **Google PageRank**

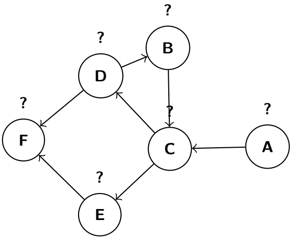
Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{n}\mathbb{1}_{n \times n}$, where $p = 0.15$

Success story #2 **Google PageRank**

Google matrix: $\mathbf{G} = (1 - p)\mathbf{M} + p \cdot \frac{1}{n}\mathbb{1}_{n \times n}$, where $p = 0.15$

G is **stochastic** why? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1.

Perron's theorem: Such v exists and it is **unique** (if the entries of **G** are positive).

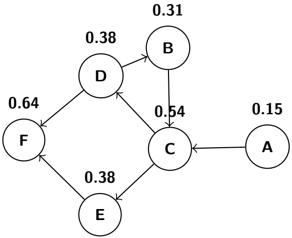


Success story #2 **Google PageRank**

Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{n}\mathbb{1}_{n \times n}$, where $p = 0.15$

G is **stochastic** why? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1.

Perron's theorem: Such v exists and it is **unique** (if the entries of **G** are positive).

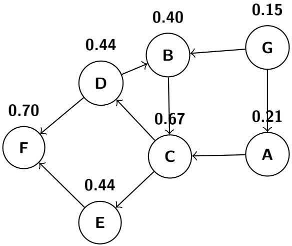


Success story #2 **Google PageRank**

Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{n}\mathbb{1}_{n\times n}$, where $p = 0.15$

G is **stochastic** why? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1.

Perron's theorem: Such v exists and it is **unique** (if the entries of **G** are positive).



Success story #2 **Google PageRank**

Problem: Find left eigenvector of a stochastic matrix.

History: [Desikan, 2006]

- The anatomy of a large-scale hypertextual web search engine [Brin & Page 1998]
- US patent for PageRank granted in 2001
- Google indexes 10's of billions of web pages (1 billion $= 10^9$)
- Google serves ≥ 200 million queries per day
- Each query processed by ≥ 1000 machines
- All search engines combined process more than 500 million queries per day

Success story #2 **Google PageRank**

Problem: Find an eigenvector of a stochastic matrix.

- $n = 10^9$!!!
- luckily: **sparse** (average outdegree: 7)
- better than a simple centrality measure (e.g., degree)
- power method

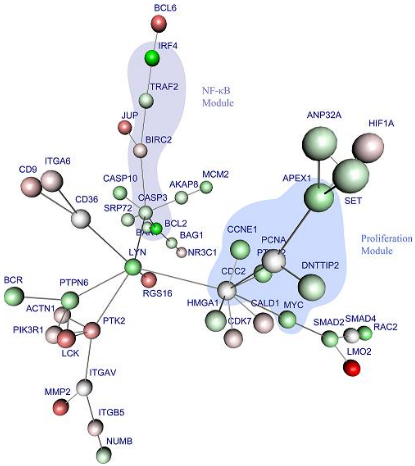
$$\mathbf{v}_0 = (1_A \quad 0_B \quad 0_C \quad 0_D \quad 0_E \quad 0_F)^\mathsf{T}$$
$$\mathbf{v}_1 = \mathbf{G}\mathbf{v}_0$$
$$\mathbf{v}_{t+1} = \mathbf{G}\mathbf{v}_t = \mathbf{G}^{t+1}\mathbf{v}$$

$\mathbf{v}_{t+1} = \mathbf{v}_t \implies \mathbf{G}\mathbf{v}_t = \mathbf{v}_t$ and we found the steady vector

But wait, **M** is sparse, but **G** is dense! What to do?

Graphs from **biological networks**

- protein-protein interactions
- gene regulatory networks
- typical ML tasks
  - discover unexplored interactions
  - learn or reconstruct the structure



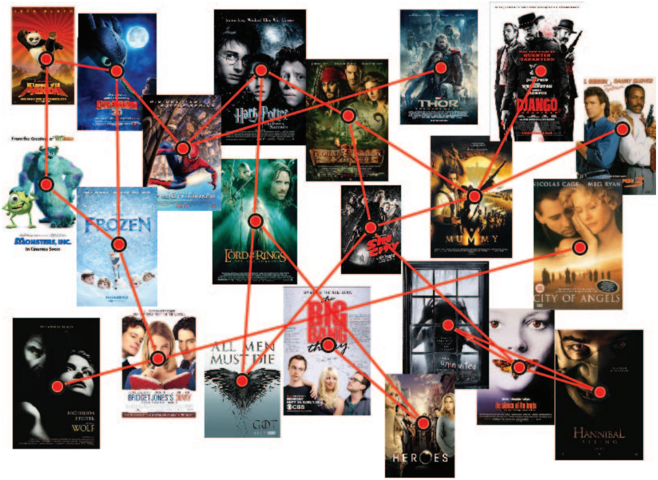
Diffuse large B-cell lymphomas - Dittrich et al. (2008)

Graphs from **similarity networks**

graph is not naturally given

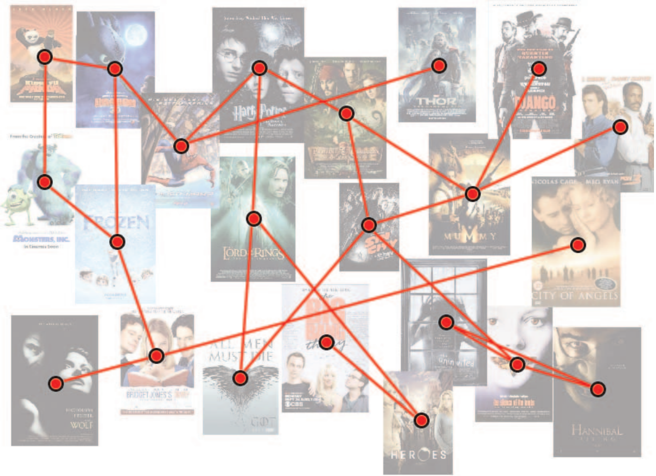
Graphs from **similarity networks**

but we can construct it



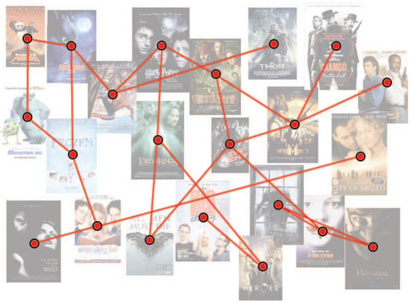
Graphs from **similarity networks**

and use it as an abstraction



Graphs from **similarity networks**

- vision
- audio
- text
- typical ML tasks
  - semi-supervised learning
  - spectral clustering
  - manifold learning



Movie similarity

Two sources of graphs in ML

Graph as models for networks

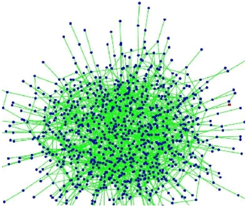
- given as an input
- discover interesting properties of the structure
- represent useful information (viral marketing)
- be the object of study (anomaly detection)

Graph as nonparametric basis

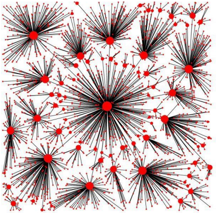
- we create (learn) the structure
- flat vectorial data $\rightarrow$ similarity graph
- nonparametric regularizer
- encode structural properties: smoothness, independence, ...

Random Graph Models

Erdős-Rényi
independent edges



Barabási-Albert
preferential attachment



Stochastic Blocks
modeling communities

Watts-Strogatz, Chung-Lu, Fiedler, ...

What will you learn in the Graphs in ML course?

Concepts, tools, and methods to work with graphs in ML.

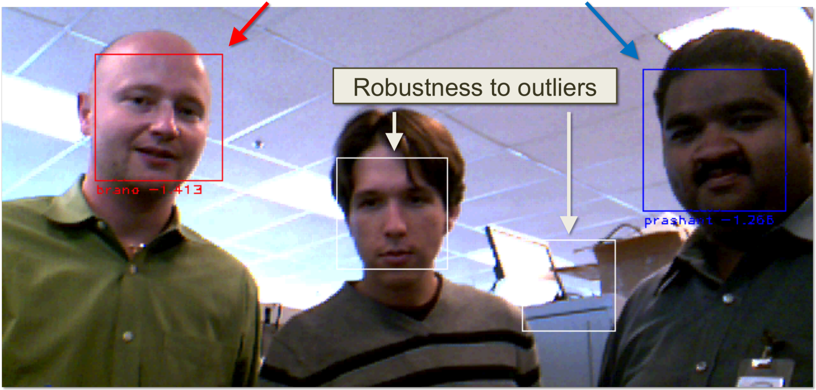
Theoretical toolbox to analyze graph based algorithms.

Specific applications of graphs in ML.

One example: Online Semi-Supervised Face Recognition

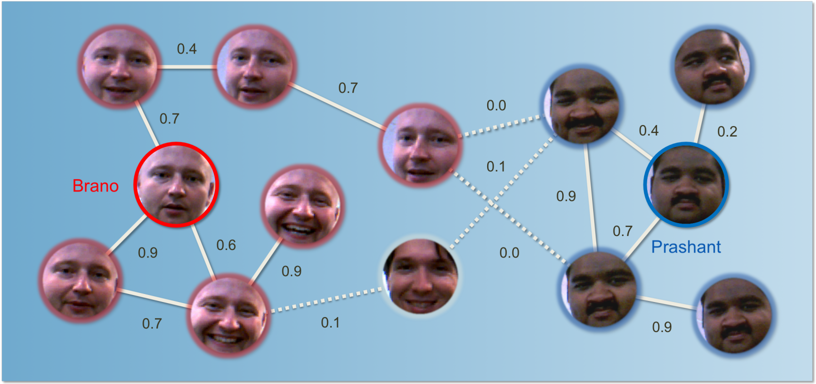
Online Semi-Supervised Face Recognition

graph is not given



Online Semi-Supervised Face Recognition

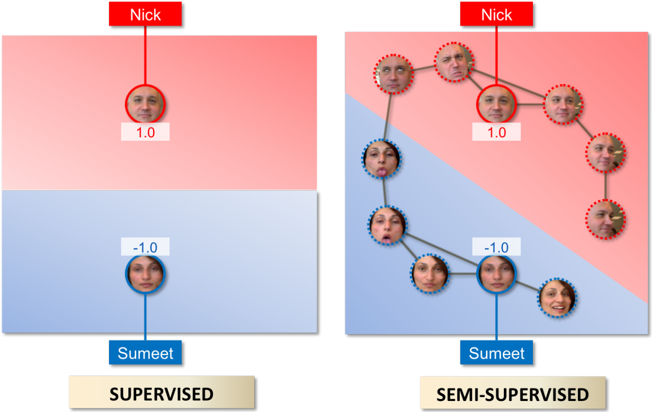
we will construct it!



An example of a similarity graph over faces. The faces are vertices of the graph. The edges of the graph connect similar faces. Labeled faces are outlined by thick solid lines.

Online Semi-Supervised Face Recognition

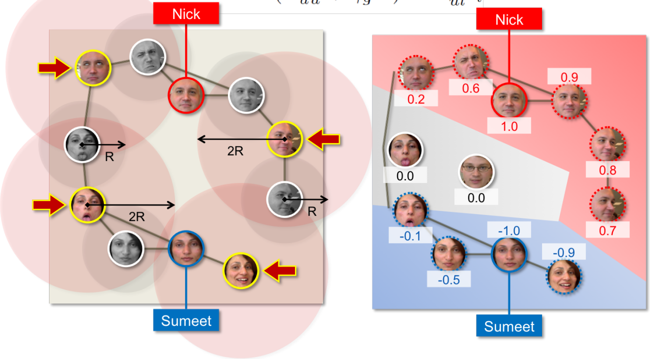
graph-based semi-supervised learning



Online Semi-Supervised Face Recognition

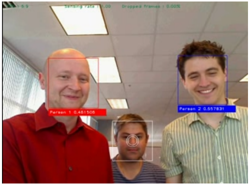
online learning - graph sparsification

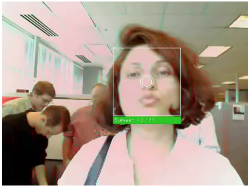
$$\ell^{q} = (L^{q}_{uu} + \gamma_g V)^{-1} W^{q}_{ul} \ell_l$$



DEMO

second TD





see the demo: `http://researchers.lille.inria.fr/~valko/hp/serve.php?what=publications/kveton2009nipsdemo.officespace.mov`

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t(\ell_t^{q}[t]-y_t)^2 \le \frac{3}{n}\sum_t(\ell_t^{*}-y_t)^2 + \frac{3}{n}\sum_t(\ell_t^{o}[t]-\ell_t^{*})^2 + \frac{3}{n}\sum_t(\ell_t^{q}[t]-\ell_t^{o}[t])^2$$

Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n_l^{3/2})$, the difference between the risks on labeled and all vertices decreases at the rate of $O(n_l^{-1/2})$ (with a high probability)

$$\frac{1}{n}\sum_t(\ell_t^{*}-y_t)^2 \le \frac{1}{n_l}\sum_{i\in l}(\ell_i^{*}-y_i)^2 + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta+4)$$

$$\beta \le \left[\frac{\sqrt{2}}{\gamma_g+1} + \sqrt{2n_l}\,\frac{1-\sqrt{c_u}}{\sqrt{c_u}}\,\frac{\lambda_M(L)+\gamma_g}{\gamma_g^2+1}\right]$$

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t (\ell_t^{\mathrm{q}}[t]-y_t)^2 \le \frac{3}{n}\sum_t (\ell_t^*-y_t)^2 + \frac{3}{n}\sum_t (\ell_t^{\mathrm{o}}[t]-\ell_t^*)^2 + \frac{3}{n}\sum_t (\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t])^2$$

Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n^{1/4})$, the average error between the offline and online HFS predictions decreases at the rate of $O(n^{-1/2})$

$$\frac{1}{n}\sum_t (\ell_t^{\mathrm{o}}[t]-\ell_t^*)^2 \le \frac{1}{n}\sum_t \left\|\ell^{\mathrm{o}}[t]-\ell^*\right\|_2^2 \le \frac{4n_l}{(\gamma_g+1)^2}$$

$$\|\ell\|_2 \le \frac{\|y\|_2}{\lambda_m(C^{-1}K+I)} = \frac{\|y\|_2}{\lambda_m(K)\lambda_M^{-1}(C)+1} \le \frac{\sqrt{n_l}}{\gamma_g+1}$$

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t (\ell_t^{\mathrm{q}}[t]-y_t)^2 \leq \frac{3}{n}\sum_t (\ell_t^{*}-y_t)^2 + \frac{3}{n}\sum_t (\ell_t^{\mathrm{o}}[t]-\ell_t^{*})^2 + \frac{3}{n}\sum_t (\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t])^2$$

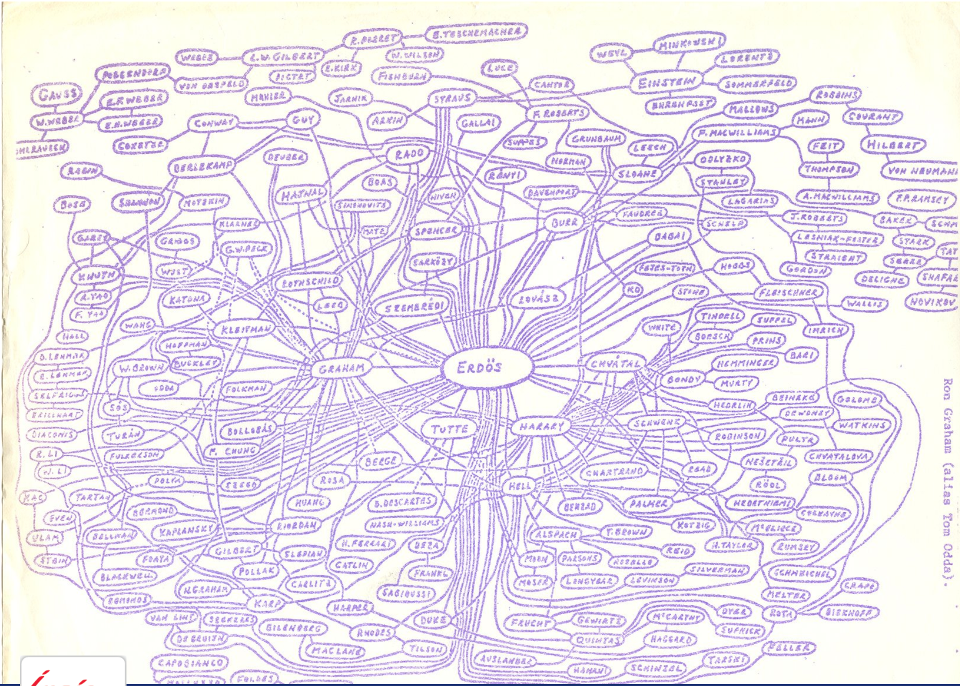
Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n^{1/8})$, and the Laplacians L^q and L^o and normalized, the average error between the online and online quantized HFS predictions decreases at the rate of $O(n^{-1/2})$

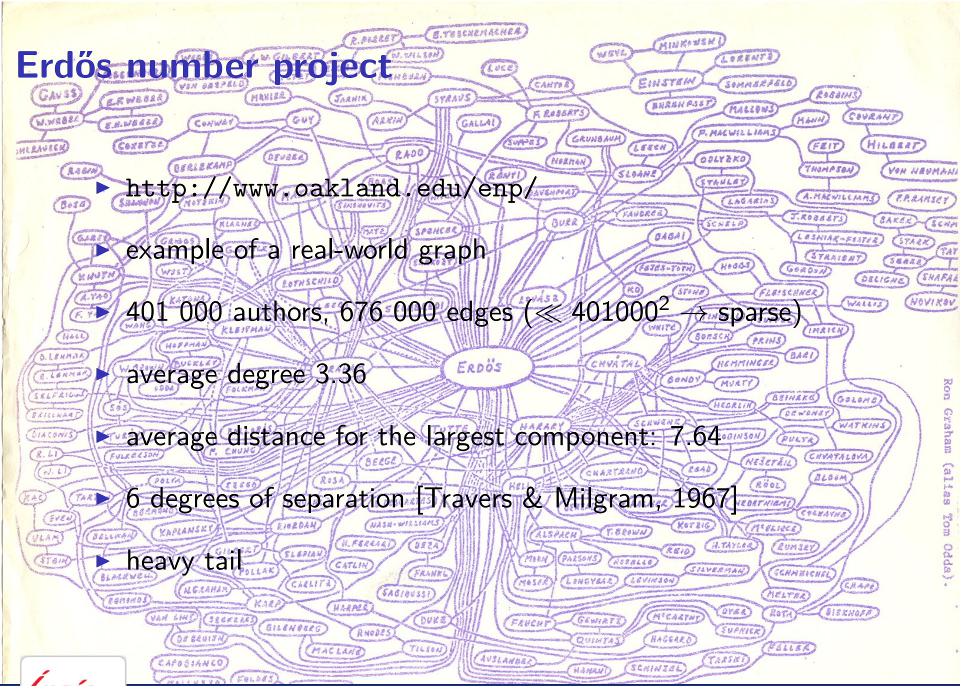
$$\frac{1}{n}\sum_t (\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t])^2 \leq \frac{1}{n}\sum_t \left\|\ell^{\mathrm{q}}[t]-\ell^{\mathrm{o}}[t]\right\|_2^2 \leq \frac{n_l}{c_u^2\gamma_g^4}\left\|L^{\mathrm{q}}-L^{\mathrm{o}}\right\|_F^2$$

$$\left\|L^{\mathrm{q}}-L^{\mathrm{o}}\right\|_F^2 \propto O(k^{-2/d})$$

The distortion rate of online k-center clustering is $O(k^{-1/d})$, where d is dimension of the manifold and k is the number of representative vertices

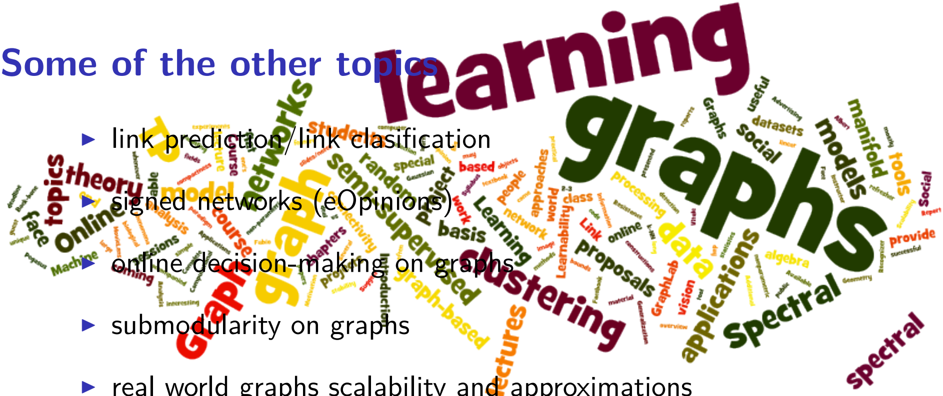

ERDÖS
GRAHAM
HARARY
TUTTE
CHVÁTAL
EINSTEIN
MINKOWSKI
LORENTZ
SOMMERFELD
WEYL
EHRENFEST
GAUSS
W.WEBER
E.H.WEBER
CONWAY
COXETER
GUY
BERLEKAMP
RADO
STRAUS
F. ROBERTS
GALLAI
GRUNBAUM
SLOANE
MALLOWS
F. MACWILLIAMS
HILBERT
VON NEUMANN
COURANT
ODLYZKO
THOMPSON
A. MACWILLIAMS
F.P.RAMSEY
J.ROBERTS
BURR
BABAI
SZEMERÉDI
LOVÁSZ
KLEITMAN
KNUTH
A. YAO
F. YAO
HAJNAL
SPENCER
SÁRKÖZY
ROTHSCHILD
KATONA
G.W.PECK
SHANNON
W. BROWN
FOLKMAN
SÓS
TURÁN
BOLLOBÁS
F. CHUNG
BERGE
ROSA
HELL
HUANG
B.DESCARTES
NASH-WILLIAMS
RIORDAN
KAPLANSKY
GILBERT
SLEPIAN
POLLAK
KARP
DUKE
HARPER
RHODES
MACLANE
SCHWENK
BONDY
MURTY
HEMMINGER
ROBINSON
CHARTRAND
PALMER
ALSPACH
T.BROWN
FRUCHT
QUINTAS
HAGGARD
McCARTHY
TARSKI
FELLER
ULAM
Ron Graham (alias Tom Odda).

Erdős number project

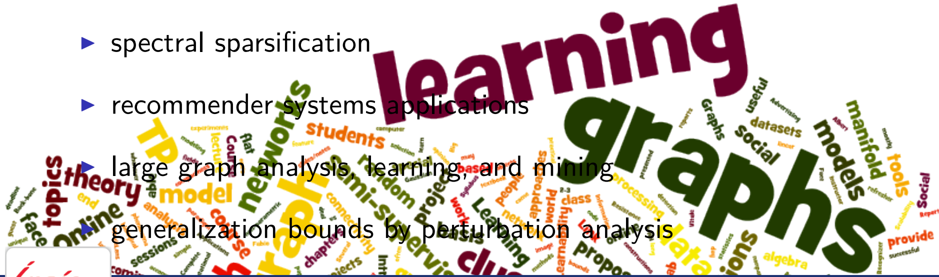


- http://www.oakland.edu/enp/
- example of a real-world graph
- 401 000 authors, 676 000 edges ($\ll 401000^2 \to$ sparse)
- average degree 3.36
- average distance for the largest component: 7.64
- 6 degrees of separation [Travers & Milgram, 1967]
- heavy tail

Some of the other topics



- link prediction/link clasification
- signed networks (eOpinions)
- online decision-making on graphs
- submodularity on graphs
- real world graphs scalability and approximations
- spectral sparsification
- recommender systems applications
- large graph analysis, learning, and mining
- generalization bounds by perturbation analysis



MVA and Graphs: 2 courses

The two MVA graph courses offer complementary material.

Fall: **Graphs in ML**

this class

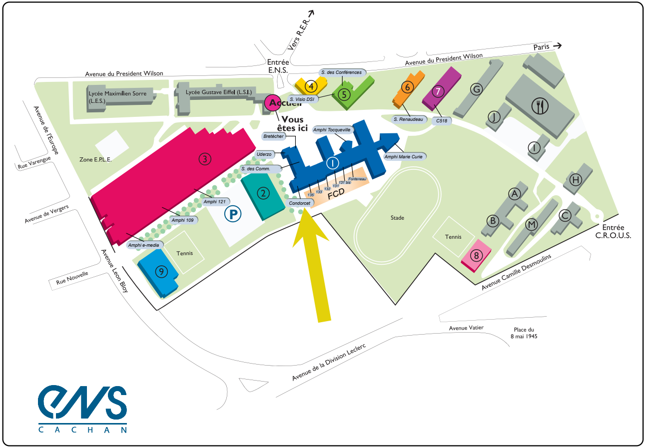
- focus on learning
- spectral clustering
- random walks
- graph Laplacian
- semi-supervised learning
- manifold learning
- theoretical analyses
- online learning
- recommender systems

Spring: **ALTeGraD**

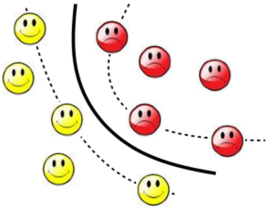
by Michalis Vazirgiannis

- dimensionality reduction
- feature selection
- text mining
- graph mining
- community mining
- graph generators
- graph evaluation measures
- privacy in graph mining
- big data

Hide-and-seek: Second class is elsewhere!



Statistical Machine Learning in Paris!



https://sites.google.com/site/smileinparis/sessions-2015--16

Speaker: Nicolò Cesa-Bianchi

Topic: Online learning with feedback graphs

Date: Monday September 28

Time: 13:30 - 14:30 (this is pretty soon)

Place: Institut Henri Poincaré — salle 314

Administrivia

Time: Mondays 11h-13h

Place: ENS Cachan - Salle Cordocet & Amphi Curie

8 lectures: 28. 9. 5.10. 12.10. 26.10 2.11. 9.11. 23.11. 30.11

3 recitations (TDs): 19.10. 16.11. 7.12.

Validation: grades from TDs (40%) + class project (60%)

Research: contact me for *internships*, *PhD. theses*, *projects*, etc.

Course website:
`http://researchers.lille.inria.fr/~valko/hp/mva-ml-graphs`

Online class discussions and announcements:
`https://piazza.com/ens_cachan/fall2015/mvagraphsml`
class code given during the class

Contact:
Lecturer: Michal.Valko @ inria.fr
TA: Daniele.Calandriello @ inria.fr

Michal Valko
michal.valko@inria.fr
sequel.lille.inria.fr